Configuração básica da rede

Sumário

Índice de tabelas

Índice de Figuras

Capítulo 1

Configuração básica da rede

1.1. Objetivos

- Configuração de interfaces de rede;
- Arquivos de configuração.

1.2. Mãos a obra

A configuração da rede em ambientes Linux, para clientes como servidores se resume em modificar arquivos de configuração, usar comandos de diagnósticos, criar e ou apagar rotas e scripts para ativar e desativar serviços e interfaces de rede.

Mas por onde eu começo?

Vamos começa pela interface de rede, em especial o cabo da placa de rede! No Linux você pode usar o comando mii-tool para verificar a conectividade de suas interfaces de rede. Vamos a prática:

 # mii-tool

```
eth0: no autonegotiation, 1000baseT-FD flow-control, link ok
```

Posso verificar que esta tudo ok, mas falta ainda a configuração do IP da placa. Neste caso posso usar o comando ifconfig, que é usado para vários tipos de configurações de uma interface de rede, como por exemplo definir um IP, mudar MAC Address, criar/excluir interfaces virtuais, entre outros. Vamos verificar todas as interfaces de redes e se temos IP.

 # ifconfig -a

```
eth0      Link encap:Ethernet  Endereço de HW 08:00:27:71:33:b0
          endereço inet6: fe80::a00:27ff:fe71:33b0/64 Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:2592 errors:0 dropped:0 overruns:0 frame:0
          TX packets:101 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:1000
          RX bytes:458485 (447.7 KiB)  TX bytes:16550 (16.1 KiB)

lo        Link encap:Loopback Local
          inet end.: 127.0.0.1  Masc:255.0.0.0
          endereço inet6: ::1/128 Escopo:Máquina
          UP LOOPBACKRUNNING  MTU:16436  Métrica:1
          RX packets:166 errors:0 dropped:0 overruns:0 frame:0
          TX packets:166 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:0
          RX bytes:19832 (19.3 KiB)  TX bytes:19832 (19.3 KiB)
```

Em nosso exemplo apenas uma interface eth0 e sem IP. Vamos usar o comando ifconfig para atribuir um ip a placa.

 # ifconfig eth0 192.168.200.10

 # ifconfig eth0

```
eth0      Link encap:Ethernet  Endereço de HW 08:00:27:71:33:b0
          inet end.: 192.168.200.10  Bcast:192.168.200.255  Masc:255.255.255.0
          endereço inet6: fe80::a00:27ff:fe71:33b0/64 Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:2722 errors:0 dropped:0 overruns:0 frame:0
          TX packets:109 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:1000
          RX bytes:486765 (475.3 KiB)  TX bytes:18680 (18.2 KiB)
```

Mas essa configuração já fica pronta no próximo boot?

Não! É necessário editar um arquivo de configuração para ser lido na inicialização do sistema. No Linux você vai encontrar vários comandos para alterar naquele momento uma configuração, e arquivos para serem lidos na inicialização do sistema.

Vamos a prática editando o arquivo /etc/network/interfaces:

 # vim /etc/network/interfaces

```
 1 # This file describes the network interfaces available on your system
 2 # and how to activate them. For more information, see interfaces(5).
 3
 4 # The loopback network interface
 5 auto lo
 6 iface lo inet loopback
 7
 8 # The primary network interface
 9 allow-hotplug eth0
10 iface eth0 inet static
11         address 192.168.200.10
12         netmask 255.255.255.0
13         network 192.168.200.0
14         broadcast 192.168.200.255
15         gateway 192.168.200.254
```

Este é arquivo de configuração das interfaces de rede no Debian.

No Debian: Para as configurações terem efeito use os comandos:

 # /etc/init.d/networking restart

ou

 # invoke-rc.d networking restart

Para você alterar IP, mascara, rede, etc no RedHat, use o comando:

 # vim /etc/sysconfig/network-scripts/ifcfg-eth0

```
# Intel Corporation 82540EM Gigabit Ethernet Controller
DEVICE=eth0
BOOTPROTO=static
BROADCAST=192.168.200.255
HWADDR=08:00:27:E2:E4:A1
IPADDR=192.168.200.10
NETMASK=255.255.255.0
NETWORK=192.168.200.0
ONBOOT=yes
```

Para definir o gateway a configuração é feita em um outro arquivo:

 # vim /etc/sysconfig/network

```
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=localhost.localdomain
GATEWAY=192.168.200.254
```

No RedHat: Para as configurações terem efeito use os comandos:

 # /etc/init.d/network restart

ou

 # service network restart

Ainda no RedHat você pode também no modo texto usar o comando:

 # system-config-network-tui

Vamos ver agora no OpenSuse!

Para editar no modo texto configurações de interface de rede use o comando:

 # vim /etc/sysconfig/network/ifcfg-eth0

```
BOOTPROTO='static'
MTU=''
REMOTE_IPADDR=''
STARTMODE='auto'
BROADCAST='192.168.200.255'
ETHTOOL_OPTIONS=''
IPADDR='192.168.200.10/24'
NAME=''
NETWORK='192.168.200.0'
USERCONTROL='no'
```

No OpenSuse: Para as configurações terem efeito use os comandos:

 # /etc/init.d/network restart

ou

 # rcnetwork restart

Ainda no OpenSuse uma outra alternativa é o uso do YAST em modo texto:

 # yast

Acesse através do teclado as opções Network Devices → Network Settings → Edit, para alterar as configurações de sua placa de rede.

```
YaST2 - lan @ server

Network Card Setup
General  Address  Hardware
 Device Type                              Configuration Name
 Ethernet                                 eth0
( ) No IP Address (for Bonding Devices)
( ) Dynamic Address  DHCP               DHCP both version 4 and 6
(x) Statically assigned IP Address
IP Address                Subnet Mask                Hostname
192.168.200.10            /24                        server.empresa.com.br
 Additional Addresses
    Alias Name|IP Address|Netmask

    [Add][Edit][Delete]

[Help]                   [Back]                   [Cancel]                   [Next]
 F1 Help  F3 Add  F9 Cancel  F10 Next
```

Voltando ao Debian você pode usar 2 comandos para desativar e ativar a configuração de um interface de rede. Vamos a prática:

Para desativar a placa eth0:

```
# ifdown eth0
```

Para ativar a placa eth0:

```
# ifup eth0
```

Os comandos só vão funcionar se a interface estiver configurada no arquivo /etc/network/interfaces

Adicionando rota

Vamos continuar nossa configuração!

Já temos a interface com conectividade, configuração pronta na inicialização, como IP, mascara e rota. Mas caso você precise alterar a rota padrão?

O comando route (independente da distribuição) pode ser usado para exibir, criar e excluir rotas para um host ou uma rede. Vamos a prática exibindo a rota atual:

```
# route -n
```

```
Tabela de Roteamento IP do Kernel
Destino         Roteador        MáscaraGen.     Opções Métrica Ref    Uso Iface
192.168.200.0   0.0.0.0         255.255.255.0   U      0       0        0 eth0
0.0.0.0         192.168.200.254 0.0.0.0         UG     0       0        0 eth0
```

Veja que nossa rota aponta para 192.168.200.254, que é o gateway da rede. Mas se o gateway mudar de IP?

Primeiro exclua a rota padrão:

```
# route del default
```

Agora adicione a nova rota:

 # route add default gw 192.168.200.30

Pronto! Sua nova rota esta configurada.

Configuração de cliente DNS

Até agora nossa configuração garante que computadores possam se comunicar através de números dentro da LAN e WAN. Mas com sei se minha configuração de DNS esta funcionado?

Primeiro ping um endereço da internet através do nome:

 # ping www.terra.com.br

```
ping: unknown host www.terra.com.br
```

Como podemos ver o sistema mostra que o host é desconhecido. Para resolver isso o Linux conta com um arquivo de configuração /etc/resolv.conf.

No resolv.conf você adiciona endereços de servidores DNS externos, e algumas opções de configuração:

domain - Define o nome do domínio local;

search - Especifica uma lista de nomes de domínio alternativos;

nameserver - Define um ou mais IPs de servidores para resolução de nomes.

Exemplo:

```
domain server.empresa.com.br
search server2.empresa.com.br
nameserver 200.144.18.16
nameserver 200.144.18.18
```

Para a resolução de nomes apenas na rede interna (LAN), o Linux conta o arquivo /etc/hosts. Através dele é feito o relacionamento entre um nome de computador e endereço IP da LAN. Vamos a prática:

 # vim /etc/hosts

```
 1 127.0.0.1        localhost.localdomain    localhost
 2 192.168.200.10   server.empresa.com.br    server
 3
 4 # The following lines are desirable for IPv6 capable hosts
 5 ::1     localhost ip6-localhost ip6-loopback
 6 fe00::0 ip6-localnet
 7 ff00::0 ip6-mcastprefix
 8 ff02::1 ip6-allnodes
 9 ff02::2 ip6-allrouters
10 ff02::3 ip6-allhosts
```

Em nosso exemplo a maquina de IP 192.168.200.10 pode ser acessada por server, server.empresa.com.br e empresa.com.br se for um servidor de DNS.

Existem uma relação entre os arquivos /etc/hosts e /etc/hostname. O nome da maquina deve estar igual nos 2 arquivos.

Para exibir ou alterar o nome da sua maquina use o comando hostname:

 # hostname

Agora para deixar padrão na inicialização, edite o arquivo /etc/hostname:

 # vim /etc/hostname

Depois de todas essas configurações podemos ainda, alterar a base de dados que o sistema vai utilizar para procurar os nomes de usuários e senhas. Isso é possível através do arquivo /etc/nsswitch.conf.

O padrão é o sistema pesquisar nomes de usuários e senhas nos arquivos /etc/passwd e /etc/shadow, mas você pode por exemplo mudar a pesquisa apontando para um servidor NIS, Ldap, AD da Microsoft, etc. Veja um exemplo:

 # vim /etc/nsswitch.conf

```
 1 # /etc/nsswitch.conf
 2 #
 3 # Example configuration of GNU Name Service Switch functionality.
 4 # If you have the `glibc-doc-reference' and `info' packages installed, try:
 5 # `info libc "Name Service Switch"' for information about this file.
 6
 7 passwd:         compat ldap
 8 group:          compat ldap
 9 shadow:         compat ldap_
10
11 hosts:          files dns
12 networks:       files
13
14 protocols:      db files
15 services:       db files
16 ethers:         db files
17 rpc:            db files
18
19 netgroup:       nis
```

Em nosso exemplo quando um usuário for autenticar na maquina, o sistema vai pesquisar seu nome no arquivo /etc/passwd e caso não encontre, vai pesquisar em uma base Ldap.

Capítulo 2

Gerenciando

2.1. Objetivos

- Troubleshooting: Modificar arquivos de configuração.

2.2. Troubleshooting

Como posso configurar o FQDN de minha maquina?

A configuração do FQDN (nome de domínio totalmente qualificado) é muito importante quando se trabalha com servidores, é através dele onde os serviços de DNS, Web (Apache) e Email (Postfix) trabalham. O FQDN é composto do nome_da_maquina+dominio.

Exemplo:

server.empresa.com.br

A configuração é feita no arquivo /etc/hosts.

 # vim /etc/hosts

```
 1 127.0.0.1       localhost.localdomain   localhost
 2 192.168.200.10  server.empresa.com.br   server
 3
 4 # The following lines are desirable for IPv6 capable hosts
 5 ::1     localhost ip6-localhost ip6-loopback
 6 fe00::0 ip6-localnet
 7 ff00::0 ip6-mcastprefix
 8 ff02::1 ip6-allnodes
 9 ff02::2 ip6-allrouters
10 ff02::3 ip6-allhosts
```

Para testar a configuração do arquivo /etc/hosts use os seguintes comandos:

Verificar o nome da maquina:

 # hostname

Verificar o IP da maquina:

```
# hostname -i
```

Verificar o domínio da maquina:

```
# hostname -d
```

Verificar o FQDN da maquina:

```
# hostname -f
```